Dependent coding in quantized matching pursuit

Vivek K Goyal and Martin Vetterli*

Dept. of Electrical Engineering and Computer Science
University of California, Berkeley
v.goyal@ieee.org, Martin.Vetterli@de.epfl.ch

ABSTRACT

Matching pursuit, introduced by Mallat and Zhang,[1] is an algorithm for decomposing a signal into a linear combination of functions chosen from possibly redundant dictionary of functions. (A similar greedy algorithm is well known for finding sparse approximate solutions to underdetermined linear systems of equations.[2,3]) A variant which we call quantized matching pursuit (QMP) has been proposed for various lossy compression problems. Here a simple dependent coding scheme is introduced to code the coefficients and indices in a quantized matching pursuit representation. The improvement in rate-distortion performance is shown through simulations on synthetic sources. The resulting system is used to code still images and motion-compensated video residual images. Since a DCT-basis dictionary is used, the multiplicative computational complexity is equal to that of traditional transform coding. The image coding results are ambiguous, with a very slight increase in PSNR but no discernible subjective improvement. The video coding results are more promising, with bit rate reductions of up to 20% comparing at constant SNR. The competitive performance and design flexibility indicate that the method warrants further investigation.

1 INTRODUCTION

Matching pursuit was introduced to the signal processing community as an algorithm for "decomposing any signal into a linear expansion of waveforms that are selected from a redundant dictionary of functions."[1] The original applications were the detection of coherent signal structures and pattern extraction from noisy signals. Since extraction of the most salient features needed for reconstruction amounts to lossy compression, many authors have proposed video[4,5] and image[6,7] coders that employ matching pursuit. These methods avoid block boundary effects because they code entire images or video frames as single units. On the other hand, they suffer from high computational complexity because they require many inner product computations and many searches over large sets.

In previous papers, we introduced quantized matching pursuit (QMP) as a technique for coding of a vector-valued source and considered the problem of optimally reconstructing from a QMP representation.[8–10] Our emphasis was on the geometric ramifications of coefficient quantization. The primary contribution of this paper is

*M. Vetterli is also with the Laboratoire de Communications Audiovisuelles, Département d'Électricité, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland.

the introduction of an efficient lossless coding technique to be used along with QMP. A QMP coding system using this dependent coding scheme can provide bounded distortion on a vector-wise basis. A secondary contribution of this paper is to suggest that QMP may be useful even when the dictionary set is an orthogonal basis. In this case, the computational complexity is rather low because coding requires a single set of inner product computations and a single sorting operation. The effectiveness of QMP comes from its prioritization of transform coefficients. For low bit-rate coding, favorable compression results have been obtained with QMP as compared to a traditional transform coding system.

The matching pursuit algorithm and its variant QMP are reviewed in Section 2. The subsequent section presents the rationale behind the new dependent coding scheme employed with QMP. Simulation results on a synthetic signal are also presented. Simulation results on coding still images and motion-compensated video residual images are presented in Section 4. The coding is block discrete cosine transform (DCT) based, and hence one could use standard DCT hardware in a hardware implementation. The final section offers conclusions and some possible directions for future work.

2 MATCHING PURSUIT AND QUANTIZED MATCHING PURSUIT

In many signal processing applications, a signal is decomposed into a linear combination of basis elements and performance depends heavily on the selection of the basis. For example, in image compression it is widely believed that the DCT basis performs well for smooth regions but very poorly for regions with many edges. Finding representations with respect to an overcomplete set (as opposed to a basis) provides flexibility to well approximate a wider range of signals.

In the finite dimensional case, once such an overcomplete set has been selected, the problem of finding an efficient representation can be abstracted as follows. Denote the signal vector by $f \in \mathbb{R}^n$. Let the columns of $A \in \mathbb{R}^{n \times m}$ contain the expansion vectors. Then Ax, $x \in \mathbb{R}^m$ is an efficient representation of b if $\|Ax - b\|_2$ is small and x has a small number of nonzero entries. With no constraints on A, finding such an x is hard:

- For given $\epsilon \in \mathbb{R}^+$, $M \in \mathbb{Z}^+$, determining if there exists x with not more than M nonzero entries such that $\|Ax - b\|_2 \leq \epsilon$ is NP-complete.[3,11]
- For given $M \in \mathbb{Z}^+$, finding x which minimizes $\|Ax - b\|_2$ among all vectors with not more than M nonzero entries is NP-hard.[11]

Matching pursuit provides a greedy, suboptimal approach to this problem. In the finite dimensional case, it is equivalent to a well-known greedy heuristic for finding sparse approximate solutions to linear equations,[2,3] without the orthogonalization step.

2.1 The matching pursuit algorithm

Let $\mathcal{D} = \{\varphi_k\}_{k=1}^M \subset \mathbb{R}^N$ span $\mathbb{R}^N$. Also impose the additional constraint that $\|\varphi_k\| = 1$ for all k. We will call $\mathcal{D}$ our *dictionary* of vectors. Matching pursuit is an algorithm to represent $f \in \mathbb{R}^N$ by a linear combination of elements of $\mathcal{D}$. Furthermore, matching pursuit is an iterative scheme that at each step attempts to approximate f as closely as possible in a greedy manner. If the dictionary is highly redundant, we expect that after a few iterations we will have an efficient approximate representation of f.

In the first step of the algorithm, k_0 is selected such that $|\langle \varphi_{k_0}, f \rangle|$ is maximized. Then f can be written as its projection onto φ_{k_0} and a residue $R_1 f$,

$$f = \langle \varphi_{k_0}, f \rangle \varphi_{k_0} + R_1 f.$$

The algorithm is iterated by treating $R_1 f$ as the vector to be best approximated by a multiple of φ_{k_1}. At step $p+1$, k_p is chosen to maximize $|\langle \varphi_{k_p}, R_p f \rangle|$ and

$$R_{p+1} f = R_p f - \langle \varphi_{k_p}, R_p f \rangle \varphi_{k_p}. \tag{1}$$

Identifying $R_0 f = f$, we can write

$$f = \sum_{i=0}^{n-1} \langle \varphi_{k_i}, R_i f \rangle \varphi_{k_i} + R_n f. \tag{2}$$

Hereafter we will denote $\langle \varphi_{k_i}, R_i f \rangle$ by α_i.

Details on the convergence of matching pursuit and other properties have been summarized well by Davis.[11] Note that the output of a matching pursuit expansion is not only the coefficients $(\alpha_0, \alpha_1, \dots)$, but also the indices $(k_0, k_1, \dots)$. For storage and transmission purposes, the indices must be accounted for.

2.2 Quantized matching pursuit

In a compression application, the coefficients must be quantized. Since only the quantized coefficient values will be available for reconstruction, using the quantized values in (1) reduces the propagation of the quantization errors. Thus, define *quantized matching pursuit* to be matching pursuit in which each coefficient is quantized prior to the calculation of the following residual. (Note that quantization destroys the orthogonality of the projection and residual.) We will denote the quantized coefficients by $\hat{\alpha}_i = q(\alpha_i)$, where q is a scalar quantization function.

In using QMP for compression, besides dictionary design, there are two factors which determine performance: the reconstruction method and the method used for lossless coding of the indices and quantized coefficients. The authors' previous work has addressed the reconstruction problem. The results are summarized in the following subsection.

2.3 Improved reconstruction using consistency

Reconstructions from QMP representations are generally computed by using the quantized coefficients in (2), giving

$$\hat{f} = \sum_{i=0}^{p-1} \hat{\alpha}_i \varphi_{k_i}. \tag{3}$$

The shortcoming of this reconstruction is that it disregards the effects of quantization; hence when the dictionary is not orthogonal it can produce inconsistent estimates. It was previously shown that a QMP representation can be viewed as a set of linear constraints, and that using (3) often gives an inconsistent estimate.[9,10] Under mild constraints on the scalar quantizers, consistent reconstructions can be found using the method of alternating projections[12] or linear programming.

The improvement due to consistent reconstruction has been confirmed experimentally. The experiments involved quantization of a zero-mean i.i.d. Gaussian source. Dictionaries were formed from sets of maximally spaced points on the unit sphere.[13] Figure 1 gives simulation results obtained with $N = 4$ and $M = 11$. Distortion is measured by mean squared-error (MSE) and rate is measured by summing the (scalar) sample entropies of

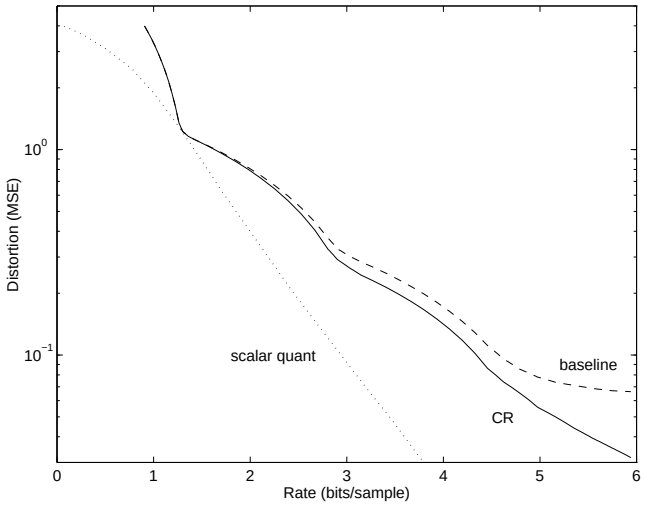


Figure 1: Distortion reduction due to consistent reconstruction. The 'baseline' method uses linear reconstruction; 'CR' refers to consistent reconstruction; 'scalar quant' refers to using independent scalar quantization for each component.

k_0, k_1, ..., k_{p-1} and $\hat{\alpha}_0$, $\hat{\alpha}_1$, ..., $\hat{\alpha}_{p-1}$, where p is the number of iterations of the algorithm. The "knees" in the curves correspond to rates at which the optimal number of iterations changes. There is no improvement at low bit rates because consistency is not an issue if there is only one iteration. As the bit rate is increased, the improvement can be dramatic.

3 RATE REDUCTION USING DEPENDENT CODING

3.1 Rationale

Even with the optimal reconstruction method of Section 2.3, the coding performance obtained was very poor. In particular, the performance was worse than that obtained with simple independent scalar quantization. Among the reasons for this poor performance are a lack of structure in the source that can be exploited in dictionary design and inefficient lossless coding. The former problem is generally difficult to address and outside the scope of this paper; we now address the latter.

For a p step expansion, the "baseline" coding method is to apply entropy codes (separately) to i_0, α_0, i_1, α_1, ..., i_{s-1}, α_{s-1}. This coding places a (rather large) penalty of roughly $\log_2 M$ bits on each iteration, *i.e.* this many bits must be spent in addition to the coding of the coefficient. In particular, the minimum achievable bit rate is about $\frac{\log_2 M}{N}$.

Assume that the same scalar quantization function is used at each iteration and that the scalar quantizer maps a symmetric interval to zero. Based on a few simple observations, we can devise a simple alternative coding method which greatly reduces the rate. The first observation is that if $\hat{\alpha}_j = 0$, then $\hat{\alpha}_k = 0$ for all $k > j$ because the residual remains unchanged. Secondly, if $\hat{\alpha}_j = 0$, then i_j carries little information. Furthermore, in this case the specification of i_j and $\hat{\alpha}_j$ defines a nonconvex set, so this information is difficult to use in reconstruction.[9]

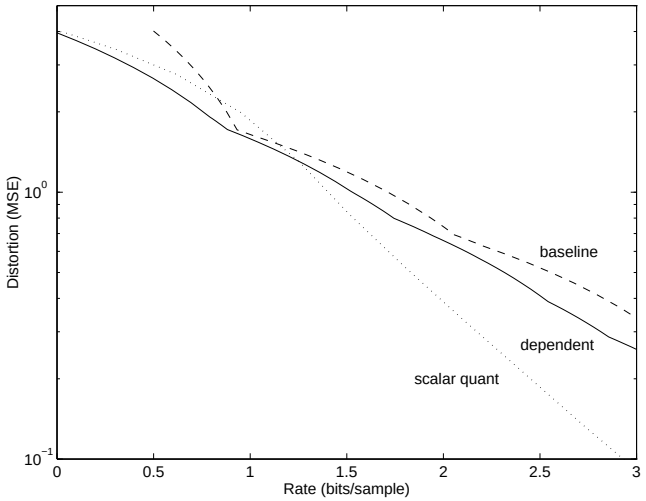


Figure 2: Simulation results with $N = 4$ showing the improvement due to dependent coding and the low bit rate superiority of QMP over independent scalar quantization.

Thus we propose the following adjustments to the coding scheme:

1. Send $\hat{\alpha}_j$ before i_j; and
2. Allow $\hat{\alpha}_j = 0$ to premature terminate a block.

We refer to this as *dependent coding*.

3.2 Simulations on synthetic signals

The improvement due to dependent coding has been assessed through experiments. Here we present the results of quantization of a zero mean i.i.d. Gaussian source. The dictionary is the standard basis. Source vectors were generated by forming blocks of N samples. Rate was measured by sample entropy and distortion by mean squared Euclidean norm. Figure 2 gives simulation results obtained with $N = 4$. The solid curve gives the performance using dependent coding, which is an improvement relative to the baseline performance given with a dashed curve. The dotted curve shows the performance of using uniform scalar quantization independently in each dimension. (Since the dictionary is orthogonal, the type of reconstruction used is not an issue.)

These simulations results are quite remarkable when suitably interpreted. Because the dictionary is an orthonormal basis, the good low bit rate performance of QMP cannot be attributed to a VQ packing gain. Since specifically the dictionary is the *standard* orthonormal basis, the performance is also not due to a transform coding gain. The favorable performance is attributable to the *prioritization* of data inherent in QMP and the variable length coding facilitated by dependent coding. One way to contrast QMP from traditional transform coding is to notice that in traditional transform coding, quantizer assignments are made based on the *expected* energies of the coefficients, while QMP allows the quantizer assignments to depend on the actual coefficient energies on a vector-by-vector basis.

These results suggest that there is a range of low bit rates for which any transform coding system can be

improved by prioritization of the coefficients. The results of an investigation of this hypothesis for DCT coding of still images and motion-compensated video residual images are given in the following section.

4 IMAGE AND VIDEO CODING RESULTS

4.1 Experimental design

Inspired by the good low bit rate coding results presented in Section 3.2, we attempted to use QMP with a DCT-basis dictionary for coding of images. (The coding method is no different for images derived from video sequences through motion compensation than it is for still images.)

QMP has a particularly simple form when the dictionary $\mathcal{D}$ is an orthogonal set. Because the quantization error $(\alpha_i - \hat{\alpha}_i)\varphi_{k_i}$ is orthogonal to $\mathcal{D} \setminus \varphi_{k_i}$, it is not necessary to find the coefficients iteratively. Instead, the inner products $\{\langle \varphi_k, f \rangle\}_{k \in \{1,2,\ldots,M\}}$ can be calculated just once (through a matrix multiplication or, in some cases, an FFT-like algorithm), and performing QMP is reduced to sorting and quantizing these inner products.

The experiments compare a "baseline" system with a QMP-based system. Both work on 8×8 pixel image blocks which have been transformed using a two dimensional separable DCT. The quantization and entropy coding are as follows:

- Baseline system: Each DCT coefficient is quantized with a uniform quantizer with step size Δ. Then each coefficient is separately entropy coded, *i.e.* there are 64 different entropy codes.
- QMP-based system: Suppose a maximum number of iterations p has been chosen. Each DCT coefficient is quantized with a uniform quantizer with step size Δ. First the largest (in absolute value) quantized coefficient $\hat{\alpha}_0$ is entropy coded. Then, if $\hat{\alpha}_0 \neq 0$, the index of the largest coefficient i_0 is entropy coded. This process continues with the next largest coefficient and terminates when either a zero quantized coefficient is reached or p coefficients and p indices have been coded. (Separate entropy codes are designed for each of $\{\hat{\alpha}_k, i_k : k = 0, 1, \ldots, p-1\}$.)

In both cases, the uniform quantizer is described by

$$x \in [(n - \frac{1}{2})\Delta, (n + \frac{1}{2})\Delta) \Rightarrow q(x) = n\Delta,\ n \in Z.$$

Instead of explicitly designing entropy codes, rate is measured by sample entropy.

4.2 Image coding results

Experiments were performed on several standard 512×512 pixel, 8-bit deep grayscale test images. To simplify the experiments, the coding of DC coefficients was ignored. Since the DC coefficient was always the largest in magnitude, the relative performances of the two systems was unaffected by this simplification. The peak signal to noise ratio (PSNR) figures and reconstructed images shown are based on exact knowledge of the DC coefficient.

Numerical results on compression of 'Lena' and 'Barbara' are shown in Figure 3. The curves were generated by first finding the lower convex hull of rate-distortion operating points and then converting the distortion measure to PSNR. For Lena, the QMP-based method gave higher PSNR for bit rates up to 0.184 bits/pixel. (Recall that this rate does not include the coding of the DC coefficient.) The peak improvement is 0.124 dB. For Barbara the

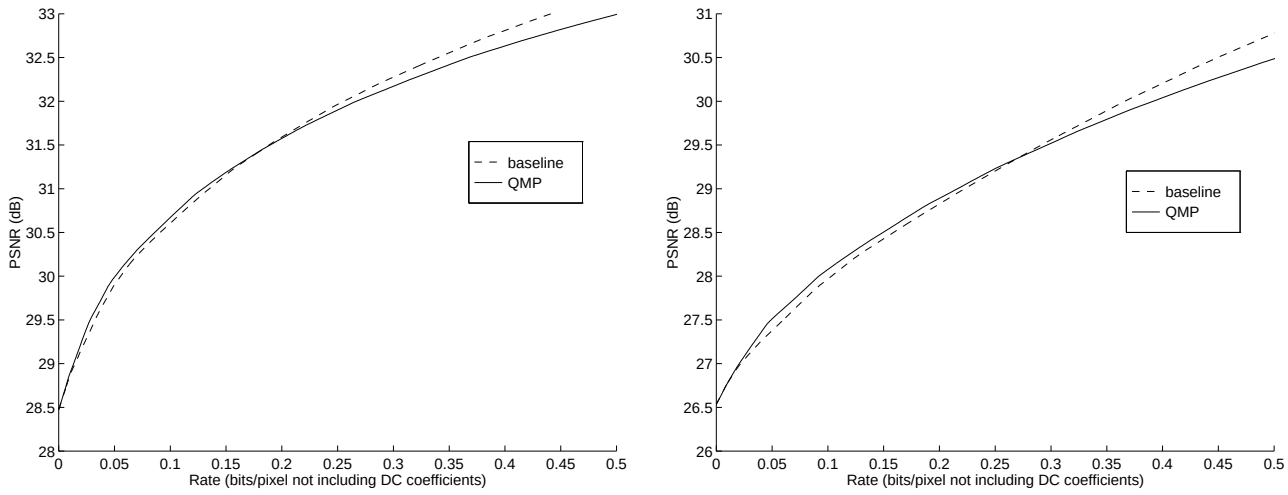


Figure 3: Simulation results for coding (a) Lena; and (b) Barbara.

QMP results are somewhat better. A peak improvement of 0.133 dB occurs at 0.050 bits/pixel and the QMP performance is better for rates up to 0.270 bits/pixel.

Figures 4 and 5 (a)–(c) allow subjective quality comparisons. The original images and the coded versions are shown at rates of 0.10 and 0.15 bits/pixel for Lena and Barbara, respectively. Despite the slightly higher PSNR obtained with QMP, the subjective quality is slightly worse.

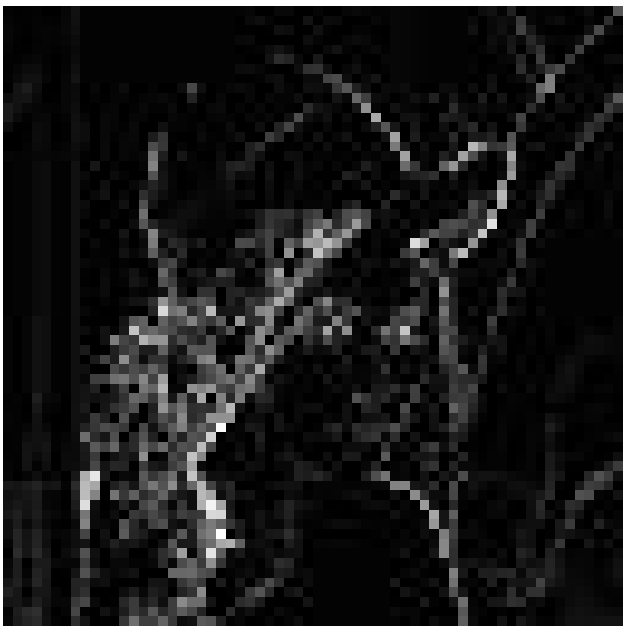
Because of the (implicit) application of entropy codes, the coded bit rate varies spatially for both coding methods. For the QMP-based method, not only does the overall bit rate vary spatially, but the number of coefficients coded varies also. For this reason it is interesting to look at the spatial rate variation for each method. This is shown in Figures 4 and 5 (d)–(e), where lighter patches correspond to more bits. For both Lena and Barbara, we find that QMP turns out to be *less* spatially adaptive than the baseline method. For Lena, we are operating at a mean rate of 6.40 bits/block with both methods. The standard deviation is 8.62 bits/block with QMP and 9.94 bits/block with the baseline method. While operating at a mean rate of 9.60 bits/block in coding Barbara, the standard deviations were 8.51 bits/block and 10.90 bits/block for the two methods, respectively.

4.3 Video coding results

Since motion-compensated video residual images are typically coded at lower rates than still images, they seem to provide an even better application for QMP coding. The two coding methods were compared for coding of four eight-frame QCIF sequences. The result are summarized in Figure 6. QMP performed better for rates up to 0.21 bits/pixel, which covers the entire range of reasonable bit rates. The peak bit savings is over 20%.

(a)

(b)



(c)

(d)

Figure 4: Results for compression of Lena at 0.10 bits/pixel (with exactly known DC coefficients): (a) coded with baseline method; (b) coded with QMP; (c) spatial rate variation with baseline method; (d) spatial rate variation with QMP.

(a) (b)

(c) (d)

Figure 5: Results for compression of Barbara at 0.15 bits/pixel (with exactly known DC coefficients): (a) coded with baseline method; (b) coded with QMP; (c) spatial rate variation with baseline method; (d) spatial rate variation with QMP.

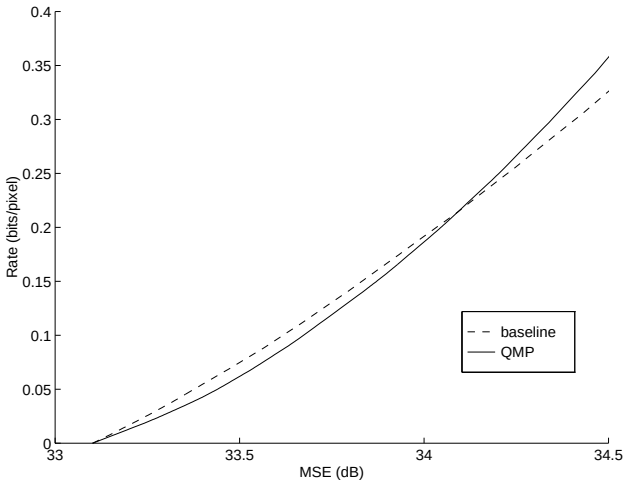


Figure 6: Simulation results for coding of four eight-frame QCIF motion-compensated video residual images.

5 INTERPRETATIONS AND FURTHER WORK

This work differs significantly from all other matching pursuit based image coding efforts of which we are aware because its complexity is only slightly higher than that of traditional DCT based coding. In particular, the multiplicative and additive complexity are the same as traditional DCT based coding because exactly the same transform coefficients are calculated.

The results obtained this far using QMP are encouraging but not spectacular. In very-low bit rate still image coding experiments, a very small PSNR improvement was obtained using QMP, but there was no accompanying subjective quality improvement. Since the relative performance of QMP is best at very low bit rates, it is probably better suited to the coding of motion-compensated video residual images. For constant MSE performance, a bit rate reduction of up to 20% was exhibited.

The baseline coding system used here for comparison differs from many DCT-based coding systems in one important way: It does not use run length coding of zeros. Nevertheless, it is not a straw man comparison because the separate entropy coding of each coefficient takes advantage of the decaying spectra of most natural images. When run length coding is used, a small nonzero coefficient sandwiched between runs of zeros has a disproportionately high cost in a rate-distortion sense. A computationally efficient, rate-distortion optimal thresholding method which exploits this effect was presented by Ramchandran and Vetterli.[14] We plan to investigate whether a similar phenomenon occurs in QMP coding, namely if the performance can be improved by sometimes widening the quantization bin at zero.

The most important result of this work is to show that a QMP coder can be competitive with a traditional transform coder at low bit rates. This fact itself could come as a surprise because of the high overhead of coding the indices of the dictionary elements used (or in the case of an orthogonal dictionary, the indices of the transform coefficients used). The flexibility of the framework allows one to experiment with using overcomplete sets for representation; because the performance is already competitive with an orthogonal dictionary, this effort does not seem misplaced.

ACKNOWLEDGEMENT

The authors would like to thank Joseph Yeh for providing the motion compensated video residual images used in Section 4.3.

5 REFERENCES

[1] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Proc.*, 41(12):3397–3415, December 1993.

[2] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins Univ. Press, Baltimore, second edition, 1989.

[3] B.K. Natarajan. Sparse approximate solutions to linear systems. *SIAM J. Computing*, 24(2):227–234, April 1995.

[4] M. Vetterli and T. Kalker. Matching pursuit for compression and application to motion compensated video coding. In *Proc. IEEE Int. Conf. Image Proc.*, volume 1, pages 725–729, Austin, TX, November 1994.

[5] R. Neff, A. Zakhor, and M. Vetterli. Very low bit rate video coding using matching pursuits. In *Proc. SPIE Conf. on Vis. Commun. and Image Proc.*, volume 2308, pages 47–60, Chicago, IL, September 1994. SPIE.

[6] F. Bergeaud and S. Mallat. Matching pursuit of images. In *Proc. IEEE Int. Conf. Image Proc.*, volume I, pages 53–56, Washington, DC, October 1995.

[7] M. Gharavi-Alkhansari and T. S. Huang. Fractal image coding using rate-distortion optimized matching pursuit. In *Proc. SPIE Conf. on Vis. Commun. and Image Proc.*, volume 2727, pages pt.3: 1386–1393, Orlando, Florida, 1996.

[8] V. K Goyal, M. Vetterli, and N. T. Thao. Quantization of overcomplete expansions. In J. A. Storer and M. Cohn, editors, *Proc. IEEE Data Compression Conf.*, pages 13–22, Snowbird, Utah, March 1995. IEEE Comp. Soc. Press.

[9] V. K Goyal and M. Vetterli. Consistency in quantized matching pursuit. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, pages 1787–1790, Atlanta, Georgia, May 1996.

[10] V. K Goyal, M. Vetterli, and N. T. Thao. Efficient representations with quantized matching pursuit. In *Proc. 12th Int. Conf. Anal. & Opt. of Sys.: Images, Wavelets & PDE's*, pages 305–311, Paris, June 1996. Springer-Verlag.

[11] G. Davis. *Adaptive Nonlinear Approximations.* PhD thesis, New York Univ., September 1994.

[12] D. C. Youla. Mathematical theory of image restoration by the method of convex projections. In H. Stark, editor, *Image Recovery: Theory and Application.* Academic Press, 1987.

[13] R. H. Hardin, N. J. A. Sloane, and W. D. Smith. Library of best ways known to us to pack n points on sphere so that minimum separation is maximized. URL: `ftp://netlib.att.com/netlib/att/math/-sloane/packings/`.

[14] K. Ramchandran and M. Vetterli. Rate-distortion optimal fast thresholding with complete JPEG/MPEG decoder compatibility. *IEEE Trans. Image Proc.*, 3(5):700–704, September 1994.